

Distributed Design Of Network Codes For Wireless Multiple Unicasts

Sukwon Kim, Michelle Effros, and Tracey Ho

Department of Electrical Engineering

California Institute of Technology

Pasadena, CA, USA 91125

Email: {sukwon, effros, tho}@caltech.edu

Abstract—Previous results on network coding for low-power wireless transmissions of multiple unicasts rely on opportunistic coding or centralized optimization to reduce the power consumption. This paper proposes a distributed strategy for reducing the power consumption in a network coded wireless network with multiple unicasts. We apply a simple network coding strategy called “reverse carpooling,” which uses only XOR and forwarding operations. In this paper, we use the rectangular grid as a simple network model and attempt to increase network coding opportunities without the overhead required for centralized design or coordination. The proposed technique designates “reverse carpooling lines” analogous to a collection of bus routes in a crowded city. Each individual unicast then chooses a route from its source to its destination independently but in a manner that maximizes the fraction of its path spent on reverse carpooling lines. Intermediate nodes apply reverse carpooling opportunistically along these routes. Our network optimization attempts to choose the reverse carpooling lines in a manner that maximizes the expected power savings with respect to the random choice of sources and sinks.

I. INTRODUCTION

By allowing coding at intermediate nodes in a network, network coding generalizes routing and obtains the maximal information flow [1]. Network coding can also be used to decrease the power consumption in wireless networks [2,3]. A simple XOR-based coding strategy is presented in [4]. Centralized strategies for applying this approach (here called “reverse carpooling”) and another simple approach called “star coding” to minimize the energy consumption for multiple unicasts appear in [5]; the complexity of that approach makes it impractical for large networks. Two polynomial-time reverse carpooling algorithms are presented in [6]; the algorithms are optimal for two and three unicasts, respectively. A greedy algorithm for the general problem also appears in [6]; while this algorithm runs in polynomial time, it requires a central controller.

In this paper, we develop a distributed strategy for reducing the expected power consumption for multiple unicasts in a network coded wireless network. Our strategy attempts to increase network coding opportunities without the overhead required for centralized design or coordination. A wireless rectangular grid is used as a simple network model. As in [7], a single node sits on each vertex of a rectangular grid, and each node can broadcast information only to its four nearest neighbors. The goal is to transmit a distinct data stream from each transmitter to its corresponding receiver in this shared

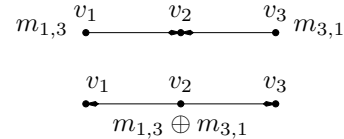


Fig. 1. Transmitting messages $m_{1,3}$ and $m_{3,1}$ from v_1 to v_3 and v_3 to v_1 , respectively, requires three transmissions with network coding and four without.

network environment. Power savings are achieved using the reverse carpooling strategy defined next.

Consider a wireless network where a node v_2 wishes to transmit message m_{13} from neighbor node v_1 to neighbor node v_3 and message m_{31} from v_3 to v_1 . This pair of messages can be sent simultaneously by transmitting the bit-wise binary sum $m_{13} + m_{31}$, as shown in Fig. 1. This strategy saves power since it halves the number of transmissions required by each intermediate node along the shared path. We call this approach *reverse carpooling* since it allows two data streams to traverse a shared path in opposite directions using approximately the same number of transmissions as it would take to send either message alone.¹

We treat distributed network code design for multiple unicasts. Like [4], our approach is opportunistic and requires no central coordination. Our strategy is to attempt to increase the number of coding opportunities by designating “reverse carpooling routes” in central locations and choosing unicast routes to maximize the fraction of the path spent on carpooling routes without increasing individual path lengths. The hope is that careful route choice will maximize the expected number of reverse carpooling opportunities. We investigate the expected power savings when multiple unicasts are chosen uniformly at random on the wireless rectangular grid.

II. PRELIMINARIES

A. System Model

We define a *rectangular grid* $G_m = (V, E)$ as the set of vertices $V = \{a(1,0) + b(0,1) : 0 \leq a, b \leq m\}$ and the set of directed edges $E = \{(v, v') : \|v - v'\| = 1\}$ where for any $v, v' \in V$, (v, v') denotes the arc connecting v

¹Precisely, in steady state the number of transmissions for reverse carpooling along a shared path is one more than the number of transmissions for one of the unicasts alone on the same path.

and v' . The head and tail of edge $e = (v_i, v_j)$ are denoted by $v_j = \text{head}(e)$ and $v_i = \text{tail}(e)$, respectively. We call the horizontal and vertical lines formed by E grid lines. A *path* is an ordered list of connected edges. Precisely, for any path $P = (e_1, e_2, \dots, e_k)$, we require $e_1, e_2, \dots, e_k \in E$ and $\text{head}(e_i) = \text{tail}(e_{i+1})$ for $1 \leq i \leq k-1$. We use $l(P) = \sum_{e \in P} \|e\| = |P|$ to denote the length of path P . For any distinct vertices $v, v' \in V$, we use $\mathcal{P}(v, v')$ to denote the set of all paths from v to v' in G_m , $\mathcal{P}^*(v, v') = \arg \min_{P \in \mathcal{P}(v, v')} l(P)$ to denote the set of the shortest paths from v to v' , and $d(v, v')$ to denote the length of the shortest path from v to v' .

In a unicast session, a single source vertex $s = (x, y) \in V$ transmits information to a single destination vertex $t = (x', y') \in V$. In this paper, we consider multiple unicast sessions on a shared rectangular grid. We specify a multiple unicast problem by describing the source and destination for each unicast. For example, $U = \{(s_1, t_1), (s_2, t_2), \dots, (s_n, t_n)\}$ is an n -unicast problem.

Given a multiple unicast problem $U = \{(s_1, t_1), (s_2, t_2), \dots, (s_n, t_n)\}$, a candidate solution $S = \{P_1, \dots, P_n\}$ is a list of paths such that $P_i \in \mathcal{P}(s_i, t_i)$ for each i . For any edge $e = (v, v') \in E$ and path $P = (e_1, e_2, \dots, e_k)$, we use $e^R = (v', v)$ and $P^R = (e_k^R, e_{k-1}^R, \dots, e_1^R)$ to denote the reversals of edge e and path P , respectively. In candidate solution S , the opportunity to apply reverse carpooling arises when two paths, say P_i and P_j , contain sub-paths $P'_i \subseteq P_i$ and $P'_j \subseteq P_j$ satisfying $(P'_i)^R = P'_j$.

The cost of a candidate solution is the energy consumed in a wireless network that transmits a single information stream along each path $P_i \in S$. When $n = 1$ (a single unicast session), we estimate the cost of a candidate solution $S = \{P_1\}$ by the number of transmissions $l(P_1)$ required to send a single packet from s_1 to t_1 along path P_1 . When $n > 1$, the opportunity for reverse carpooling may arise. We approximate the cost saved using reverse carpooling by counting the link between nodes v and v' only once for each time we apply reverse carpooling along (v, v') and (v', v) . The number of reverse carpooling opportunities along edge e is

$$\min \left\{ \sum_{P \in S} 1(e \in P), \sum_{P \in S} 1(e^R \in P) \right\},$$

and thus the resulting cost across edge e and e^R using candidate solution S is approximated as

$$C(S, e, e^R) = \max \left\{ \sum_{P \in S} 1(e \in P), \sum_{P \in S} 1(e^R \in P) \right\},$$

giving a total cost

$$C(S) = \frac{1}{2} \sum_{e \in E} \{C(S, e, e^R)\}.$$

B. Strategy

In Section III, for each network model, we design a reverse carpooling edge set $E_1 \subseteq E$. Together, the edges in E_1 form reverse carpooling routes. In Section III-A, we design E_1 so that each reverse carpooling route is a horizontal grid

line. We call this reverse carpooling route a row reverse carpooling line and this network model a row model. We begin by proposing a distributed route choice algorithm for an arbitrary row model and analyzing the edge use distribution of our algorithm. We then optimize the reverse carpooling line placement to minimize the resulting expected cost. In Section IV, we design E_1 to contain both horizontal and vertical grid lines. We again propose an algorithm, analyze the resulting edge use distribution, and optimize the line choice. In Section V we conclude with a short discussion.

III. ROW MODELS

The optimal configuration of the reverse carpooling lines may depend on factors like the size of the network, the number of unicasts, the distribution on unicasts, etc. We assume that n unicasts $U = \{(s_1, t_1), \dots, (s_n, t_n)\}$ are chosen uniformly at random on the wireless rectangular grid G_m and begin by exploring simple row models. Given a wireless rectangular grid G_m , we use a t -tuple $(0 \leq h_1 < h_2 < \dots < h_t \leq m)$ to denote the locations of t row reverse carpooling lines. (For convenience, $h_0 = 0$ and $h_{t+1} = m + 1$.) We define the reverse carpooling edge set $E_1 = \{((i, h_j), (i + 1, h_j)), ((i + 1, h_j), (i, h_j)) : 0 \leq i \leq n - 1, 1 \leq j \leq t\}$. Then edges in E_1 form t row reverse carpooling lines.

A. Path Choice Algorithm

The proposed algorithm finds a shortest path $P_i \in \mathcal{P}^*(s_i, t_i)$ that maximizes the fraction of the path spent on the row reverse carpooling lines. Let $s_i = (s_{ix}, s_{iy})$ and $t_i = (t_{ix}, t_{iy})$ and choose $0 \leq p, q \leq t$ so that $h_p \leq s_{iy} < h_{p+1}$ and $h_q \leq t_{iy} < h_{q+1}$.

Case 1) $p = q$. Here P_i is the unique path in $\mathcal{P}^*(s_i, u_i) \times \mathcal{P}^*(u_i, t_i)$ where $u_i = (t_{ix}, s_{iy})$.

Case 2) $p < q$. Here P_i is the unique path in $\mathcal{P}^*(s_i, v_i) \times \mathcal{P}^*(v_i, w_i) \times \mathcal{P}^*(w_i, t_i)$, where $v_i = (s_{ix}, h_{p+1})$ and $w_i = (t_{ix}, h_{p+1})$.

Case 3) $p > q$. Here P_i is the unique path in $\mathcal{P}^*(s_i, x_i) \times \mathcal{P}^*(x_i, y_i) \times \mathcal{P}^*(y_i, t_i)$, where $x_i = (s_{ix}, h_p)$ and $y_i = (t_{ix}, h_p)$.

B. Edge Use Distribution

Together, the edge set and path choice strategy impose a traffic distribution $r^i(e)$ for each $e \in E$ where $r^i(e)$ is the probability that $e \in P_i$. Since each unicast session is chosen uniformly at random and follows the same strategy to determine a path, $r^i(e) = r^1(e)$ for all $1 \leq i \leq n$. To obtain $r^1(e)$, we calculate the fraction of possible unicasts $(s_1, t_1) \in V^2$ for which $e \in P_1$. Fix $0 \leq p, q \leq t$ so that $h_p \leq s_{1y} < h_{p+1}$, and $h_q \leq t_{1y} < h_{q+1}$.

Case 1) $e = ((a, b), (a + 1, b))$, $e \notin E_1$. Since $e \notin E_1$, $h_i < b < h_{i+1}$ for some $0 \leq i \leq t$. Thus, $e \in P_1$ if and only if $0 \leq s_{1x} \leq a$, $s_{1y} = b$, $a + 1 \leq t_{1x} \leq m$, and $h_i \leq t_{1y} < h_{i+1}$. Therefore, $r^1(e) = \frac{(a+1)(m-a)(h_{i+1}-h_i)}{(m+1)^4}$.

Case 2) $e = ((a, b), (a + 1, b))$ and $e \in E_1$. Since $e \in E_1$, $b = h_i$ for some $1 \leq i \leq t$. If $h_{i-1} \leq s_{1y} < h_i$, then $e \in P_1$ if and only if $0 \leq s_{1x} \leq a$, $a + 1 \leq t_{1x} \leq m$,

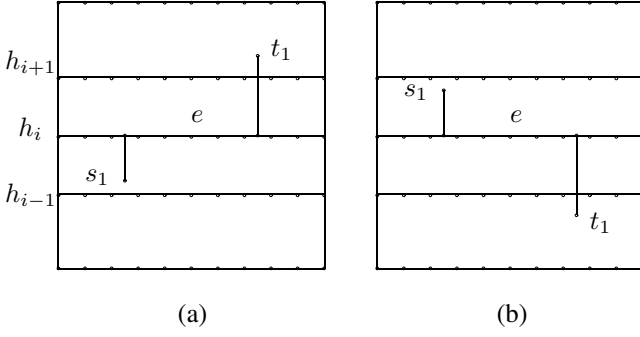


Fig. 2. Case 2 in the calculation of edge use distribution $r^1(e)$. (a) $h_{i-1} \leq s_{1y} < h_i$ and (b) $h_i < s_{1y} < h_{i+1}$.

and $h_i \leq t_{1y} \leq m$, as shown in Fig. 2(a). If $s_{1y} = h_i$, then $e \in P_1$ if and only if $0 \leq s_{1x} \leq a$, $a+1 \leq t_{1x} \leq m$, and $0 \leq t_{1y} < h_{i+1}$. If $h_i < s_{1y} < h_{i+1}$, then $e \in P_1$ if and only if $0 \leq s_{1x} \leq a$, $a+1 \leq t_{1x} \leq m$, and $0 \leq t_{1y} < h_i$, as shown in Fig. 2(b). If $s_{1y} < h_{i-1}$ or $s_{1y} \geq h_{i+1}$, then $e \notin P_1$. Thus,

$$r^1(e) = \left[\frac{(a+1)(m-a)(h_i - h_{i-1})(m+1 - h_i)}{(m+1)^4} + \frac{(a+1)(m-a)(h_{i+1} - h_i)(h_i + 1)}{(m+1)^4} \right].$$

Case 3) $e = ((a+1, b), (a, b))$. By the symmetry of our algorithm, $r^1((a+1, b), (a, b)) = r^1((m-a-1, b), (m-a, b))$. By cases 1 and 2 above, $r^1((m-a-1, b), (m-a, b)) = r^1((a, b), (a+1, b))$. Therefore, $r^1((a+1, b), (a, b)) = r^1((a, b), (a+1, b))$.

Case 4) $e = ((a, b), (a, b+1))$. Fix $0 \leq i \leq t$ so that $h_i \leq b < h_{i+1}$. In this case, $e \in P_1$ only if $0 \leq s_{1y} \leq b$ and $b+1 \leq t_{1y} \leq m$. If $0 \leq s_{1y} < h_i$, then $e \in P_1$ if and only if $0 \leq s_{1x} \leq m$, $t_{1x} = a$, and $b+1 \leq t_{1y} \leq m$, as shown in Fig. 3(a). If $h_i \leq s_{1y} \leq b$ and $h_{i+1} \leq t_{1y} \leq m$, then $e \in P_1$ if and only if $s_{1x} = a$, $0 \leq t_{1x} \leq m$, as shown in Fig. 3(b). If $h_i \leq s_{1y} \leq b$ and $b+1 \leq t_{1y} < h_{i+1}$, then $e \in P_1$ if and only if $0 \leq s_{1x} \leq m$, $t_{1x} = a$.

$$\begin{aligned} r^1(e) &= \frac{(b+1)(m-b)}{(m+1)^4} \left[\frac{h_i(m+1)}{(b+1)} + \frac{(b+1-h_i)}{(b+1)} \left(\frac{(m+1-h_{i+1})(m+1)}{(m-b)} + \frac{(m+1)(h_{i+1}-b-1)}{(m-b)} \right) \right] \\ &= \frac{(b+1)(m-b)}{(m+1)^3}. \end{aligned}$$

Case 5) $e = ((a, b+1), (a, b))$. Fix $0 \leq i \leq t$ so that $h_i \leq b < h_{i+1}$. In this case, $e \in P_1$ only if $b+1 \leq s_{1y} \leq m$ and $0 \leq t_{1y} \leq b$. If $h_{i+1} \leq s_{1y} \leq m$, then $e \in P_1$ if and only if $0 \leq s_{1x} \leq m$, $t_{1x} = a$, and $0 \leq t_{1y} \leq b$, as shown in Fig. 4(a). If $b+1 \leq s_{1y} < h_{i+1}$ and $0 \leq t_{1y} < h_i$, then

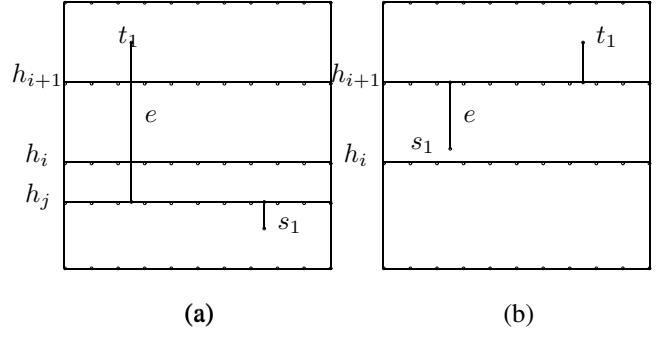


Fig. 3. Case 4 in the calculation of $r^1(e)$. (a) $0 \leq s_{1y} < h_i$ ($j \leq i$) and (b) $h_i \leq s_{1y} \leq b$ and $h_{i+1} \leq t_{1y} \leq m$.

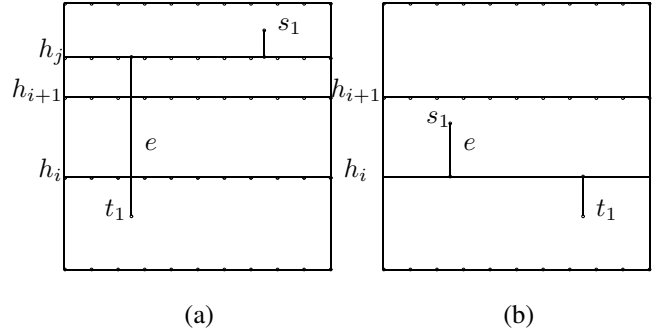


Fig. 4. Case 5 in the calculation of $r^1(e)$. (a) $h_{i+1} \leq s_{1y} \leq m$ ($j \geq i+1$) and (b) $b+1 \leq s_{1y} < h_{i+1}$ and $0 \leq t_{1y} < h_i$.

$e \in P_1$ if and only if $s_{1x} = a$, $0 \leq t_{1x} \leq m$, as shown in Fig. 4(b). If $b+1 \leq s_{1y} < h_{i+1}$ and $h_i \leq t_{1y} \leq b$, then $e \in P_1$ if and only if $0 \leq s_{1x} \leq m$, $t_{1x} = a$.

$$\begin{aligned} r^1(e) &= \frac{(m-b)(b+1)}{(m+1)^4} \left[\frac{(m+1-h_i)(m+1)}{(m-b)} + \frac{(h_{i+1}-b-1)}{(m-b)} \left(\frac{h_i(m+1)}{(b+1)} + \frac{(b+1-h_i)(m+1)}{(b+1)} \right) \right] \\ &= \frac{(b+1)(m-b)}{(m+1)^3}. \end{aligned}$$

C. Expected Network Cost

We compute the expected network cost for the row model when n unicasts $U = \{(s_1, t_1), \dots, (s_n, t_n)\}$ are chosen uniformly at random on G_m . We use S to denote the candidate solution for U obtained by our strategy and $t(n, i, j, e)$ to denote the probability that i unicasts traverse e and j unicasts traverse e^R ($0 \leq i+j \leq n$). No unicast can contain both e and e^R in its path using our path choice algorithm. Thus each unicast either uses edge e (with probability $r^1(e)$), uses

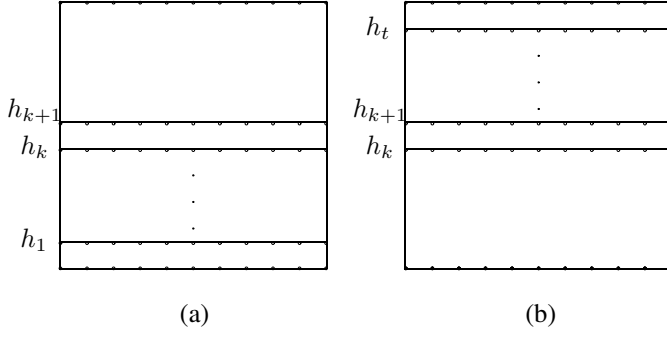


Fig. 5. Given h_k and h_{k+1} , optimizing (h_1, \dots, h_t) is equivalent to optimizing (h_1, \dots, h_{k-1}) in (a) and (h_{k+2}, \dots, h_t) in (b).

edge e^R (with probability $r^1(e) = r^1(e^R)$), or uses neither (with probability $1 - r^1(e) - r^1(e^R) = 1 - 2r^1(e)$). Thus

$$t(n, i, j, e) = \binom{n}{i} \binom{n-i}{j} (r^1(e))^{i+j} (1 - 2r^1(e))^{n-i-j}.$$

We compute the expected network cost $EC(S)$ as follows. For any (i, k) satisfying $0 \leq 2k + i \leq n$, $k + i$ unicasts use e and k unicasts use e^R with probability $t(n, k + i, k, e)$. Likewise, k unicasts use e and $k + i$ unicasts use e^R with probability $t(n, k, k + i, e) = t(n, k + i, k, e)$. In both cases, $C(S, e, e^R) = k + i$. Since we considered $i = 0$ in both cases, by the definition of the network cost, we obtain

$$\begin{aligned} EC(S) &= \sum_{e \in E} \frac{1}{2} EC(S, e, e^R) \\ &= \sum_{e \in E} \frac{1}{2} \left[\sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} \left(2 \sum_{i=0}^{n-2k} (k+i) t(n, k, k+i, e) \right) \right. \\ &\quad \left. - k \cdot t(n, k, k, e) \right]. \end{aligned} \quad (1)$$

D. Results

In this section, we first present a low-complexity algorithm that optimizes the reverse carpooling line placement to minimize the expected cost given a number of unicasts. Then, we demonstrate the performance of our algorithm.

Our goal is to optimize the locations of t reverse carpooling lines ($0 \leq h_1 < h_2 < \dots < h_t \leq m$) for n unicasts on G_m . We use E_h and E_v to denote the sets of all horizontal and vertical edges, respectively. Since the edge use distribution of any vertical edge is independent of (h_1, \dots, h_t) , from (1), $EC(S) = f(h_1, \dots, h_t) + M$ where $f(h_1, \dots, h_t) = \sum_{e \in E_h} \frac{1}{2} EC(S, e, e^R)$ and $M = \sum_{e \in E_v} \frac{1}{2} EC(S, e, e^R)$. Finding (h_1, \dots, h_t) to minimize $f(h_1, \dots, h_t)$ minimizes $EC(S)$. The following algorithm finds this optimal (h_1, \dots, h_t) by recursively dividing the problem into smaller and smaller independent subproblems.

Fix $q \geq 1$. Let $t = 2^{q+1} - 2$ and $r = 2^q - 1$. For convenience, $h_{-1} = h_0 = 0$ and $h_{t+1} = h_{t+2} = m + 1$.²

²We include h_{-1} , h_0 , h_{t+1} , and h_{t+2} in this characterization for symmetry, as will become clear in the following discussion.

Suppose that $k = \frac{t}{2}$ and the locations of the k th and $(k + 1)$ th reverse carpooling lines are given by $h_k = i_1$ and $h_{k+1} = i_1 + d_1$ ($d_1 \geq 1$). Since $r^1(e)$ for each $e \in E$ depends on at most three closest reverse carpooling lines, $r^1(e)$ is a function of either (h_1, \dots, h_{k+1}) or (h_k, \dots, h_t) for each $e \in E$. Thus, given h_k and h_{k+1} , the objective function f can be decomposed as

$$f(h_1, \dots, h_t) = f_1^{(1)}(h_{-1}, \dots, h_{k+1}) + f_2^{(1)}(h_k, \dots, h_{t+2}),$$

where functions $f_1^{(1)}$ and $f_2^{(1)}$ are independent when h_k and h_{k+1} are fixed. Here

$$f_1^{(1)}(0, 0, h_1, \dots, h_{k+1}) = \sum_{e \in E_1^{(1)}} \frac{1}{2} EC(S, e, e^R) \quad \text{and}$$

$$f_2^{(1)}(h_k, \dots, h_t, m + 1, m + 1) = \sum_{e \in E_2^{(1)}} \frac{1}{2} EC(S, e, e^R)$$

where $E_1^{(1)} = \{((a, b), (a + 1, b)), ((a + 1, b), (a, b)) : 0 \leq a < m, 0 \leq b < h_{k+1}\}$ and $E_2^{(1)} = \{((a, b), (a + 1, b)), ((a + 1, b), (a, b)) : 0 \leq a < m, h_{k+1} \leq b \leq m\}$.

The given formulation breaks our optimization problem into two subproblems. The first subproblem contains $k + 1$ reverse carpooling lines ($0 \leq h_1 < h_2 < \dots < h_{k+1} \leq m$) with $h_k = i_1$ and $h_{k+1} = i_1 + d_1$, as shown in Fig. 5(a). The goal here is to choose (h_1, \dots, h_{k-1}) to minimize $f_1^{(1)}(0, 0, h_1, \dots, h_{k-1}, i_1, i_1 + d_1)$. The second subproblem contains $k + 1$ reverse carpooling lines ($0 \leq h_k < h_{k+1} < \dots < h_t \leq m$) with $h_k = i_1$ and $h_{k+1} = i_1 + d_1$, as shown in Fig. 5(b). The goal here is to choose (h_{k+2}, \dots, h_t) to minimize $f_2^{(1)}(i_1, i_1 + d_1, h_{k+2}, \dots, h_t, m + 1, m + 1)$. The added parameters $h_{-1} = h_0 = 0$ and $h_{t+1} = h_{t+2} = m + 1$ are included so that each subproblem is bounded above and below by two reverse carpooling lines. Searching over all possible values of i_1 and d_1 and then optimizing $f_1^{(1)}(0, 0, h_1, \dots, h_{k-1}, i_1, i_1 + d_1)$ and $f_2^{(1)}(i_1, i_1 + d_1, h_{k+2}, \dots, h_t, m + 1, m + 1)$ guarantees the optimal solution.

To optimize $f_1^{(1)}(0, 0, h_1, \dots, h_{k-1}, i_1, i_1 + d_1)$ and $f_2^{(1)}(i_1, i_1 + d_1, h_{k+2}, \dots, h_t, m + 1, m + 1)$, we again apply the same approach – first fixing the two central line locations and then breaking each problem into two independent subproblems

$$\begin{aligned} &f_1^{(1)}(0, 0, h_1, \dots, h_{k-1}, i_1, i_1 + d_1) \\ &= f_1^{(2)}(0, 0, h_1, \dots, h_{l-1}, i_2, i_2 + d_2) \\ &\quad + f_2^{(2)}(i_2, i_2 + d_2, \dots, i_1, i_1 + d_1), \\ &f_2^{(1)}(i_1, i_1 + d_1, h_{k+2}, \dots, m + 1) \\ &= f_3^{(2)}(i_1, i_1 + d_1, h_{k+2}, \dots, h_{k+l-2}, i_3, i_3 + d_3) \\ &\quad + f_4^{(2)}(i_3, i_3 + d_3, h_{k+l+1}, \dots, m + 1). \end{aligned}$$

Function $l_q(a, b, c, d)$ shown in Fig. 6 captures the recursive approach. Running $l_q(a, b, c, d)$ finds the optimal $2^{q+1} - 2$ reverse carpooling lines between two upper reverse carpooling

Function $l_q(a, b, c, d)$

if $i = 0$

return $l_0(a, b, c, d) = \sum_{e \in E_{b,d}} \frac{1}{2} EC(S, e, e^R)$ (2)
 where $E_{b,d} = \{((j, k), (j+1, k)), ((j+1, k), (j, k)) : 0 \leq j < m, b \leq k < d\}$ and the expected cost $EC(S, e, e^R)$ for each $e \in E_{b,d}$ is calculated assuming reverse carpooling lines only at locations a, b, c , and d .

else

return $\min_{(x,y)} [l_{i-1}(a, b, x, y) + l_{i-1}(x, y, c, d)]$
 over all (x, y) s.t. $b + 2^i - 2 < x < y < c - 2^i + 2$.

Fig. 6. Function $l_q(a, b, c, d)$ finds the optimal $2^{q+1} - 2$ reverse carpooling lines between two upper reverse carpooling lines at locations c and d and two lower reverse carpooling lines at locations a and b and returns its expected cost.

TABLE I

OPTIMAL REVERSE CARPOOLING LINES PLACEMENT ON THE G_{10} .

n	t^*	(h_1^*, \dots, h_t^*)
$n < 55$	2	(3,7)
$n \geq 55$	3	(2,5,8)

lines at locations c and d and two lower reverse carpooling lines at locations a and b and returns its expected cost.

Theorem 1: When n unicasts are chosen uniformly at random on G_m , $l_q(0, 0, m+1, m+1)$ finds the optimal locations for $t = 2^{q+1} - 2$ row reverse carpooling lines in time $O(qm^6 + n^2m^6)$.

Proof: The optimality of our algorithm follows immediately from its search of all possible line placements. The run-time relies on the storage of all intermediate values $l_i(a, b, c, d)$ used in calculating $l_q(0, 0, m+1, m+1)$; since many of these values are used repeatedly, we avoid repeated computation by keeping a table of known values and calling the function only when the value is unknown. We calculate the run time as follows. For each $1 \leq i \leq q$ and each needed (a, b, c, d) , we find $l_i(a, b, c, d)$ as $l_{i-1}(a, b, x, y) + l_{i-1}(x, y, c, d)$ for the optimal choice (x, y) of the two central carpooling line locations. Since there are q values of i , $O(m^4)$ values of (a, b, c, d) , and $O(m^2)$ values for (x, y) , these calculations run in time $O(qm^6)$. From (1) and (2), calculation of $l_0(a, b, c, d)$ for each (a, b, c, d) runs in time $O(m^2n^2)$, giving total run-time $O(qm^6 + n^2m^6)$. ■

Tables 1 and 2 show the optimal number of reverse carpooling lines (t^*) and their optimal locations ($h_1^*, \dots, h_{t^*}^*$) for n unicasts chosen uniformly at random on G_{10} and G_{12} , respectively.

Fig. 7 plots the normalized cost $EC(S^*)/ED(U)$ as a

TABLE II

OPTIMAL REVERSE CARPOOLING LINES PLACEMENT ON THE G_{12} .

n	t^*	$(h_1^*, \dots, h_{t^*}^*)$
$n < 40$	2	(4,8)
$40 \leq n < 110$	3	(3,6,9)
$n \geq 110$	3	(2,5,9)

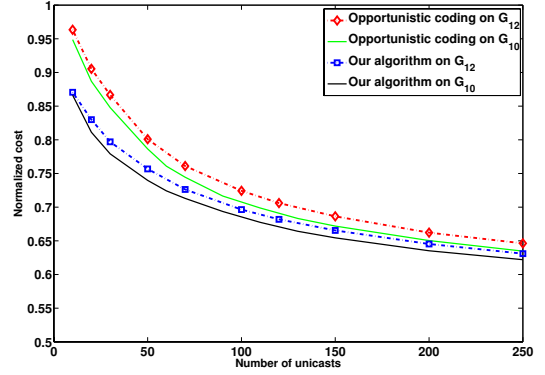


Fig. 7. Normalized cost on G_{10} and G_{12} .

function of the number of unicasts, n , where S^* is the solution given by our algorithm and $ED(U) = E \sum_{i=1}^n d(s_i, t_i)$ is the expected distance between sources and sinks for the unicasts $U = \{(s_1, t_1), \dots, (s_n, t_n)\}$. (See Lemma 1 in the Appendix.) In both cases, the normalized cost decreases as the number of unicast sessions increases. Also included are the corresponding normalized costs when no reverse carpooling lines are included and pure opportunistic coding is employed. Our algorithm yields as much as 7% improvement over pure opportunistic coding when $n < 40$ in both cases. When n is large, traffic is sufficiently large that reverse carpooling opportunities arise even without the introduction of reverse carpooling lines. As a result, the percentage improvement over opportunistic coding decreases as n increases.

IV. ROW AND COLUMN MODEL

To increase the opportunities to apply reverse carpooling, we next add column reverse carpooling lines to the previous model. Given a wireless rectangular grid G_m , we use a t -tuple $(0 \leq h_1 < h_2 < \dots < h_t \leq m)$ and a k -tuple $(0 \leq r_1 < r_2 < \dots < r_k \leq m)$ to denote the locations of t row and k column reverse carpooling lines, respectively. (For convenience, $h_0 = r_0 = 0$ and $h_{t+1} = r_{k+1} = m+1$.) The reverse carpooling edge set is $E_1 = \{((i, h_j), (i+1, h_j)), ((i+1, h_j), (i, h_j)), ((r_p, i), (r_p, i+1)), ((r_p, i+1), (r_p, i)) : 0 \leq i \leq m-1, 1 \leq j \leq t, 1 \leq p \leq k\}$.

A. Path Choice Algorithm

The proposed algorithm finds a shortest path $P_i \in \mathcal{P}^*(s_i, t_i)$ that maximizes the fraction of the path spent on the reverse carpooling lines. Choose $0 \leq c, d \leq k$ and $0 \leq f, g \leq t$ so that $r_c \leq s_{ix} < r_{c+1}$, $h_f \leq s_{iy} < h_{f+1}$, $r_d \leq t_{ix} < r_{d+1}$, and $h_g \leq t_{iy} < h_{g+1}$.

Case 1) $c = d$ or $f = g$. P_i is the unique path in $\mathcal{P}^*(s_i, u_i) \times \mathcal{P}^*(u_i, t_i)$ where $u_i = (t_{ix}, s_{iy})$.

Case 2) $c < d$ and $f \neq g$. If $f > g$, P_i is the unique path in $\mathcal{P}^*(s_i, v_i) \times \mathcal{P}^*(v_i, w_i) \times \mathcal{P}^*(w_i, x_i) \times \mathcal{P}^*(x_i, t_i)$ where $v_i = (r_{c+1}, s_{iy})$, $w_i = (r_{c+1}, h_{g+1})$, and $x_i = (t_{ix}, h_{g+1})$, as shown in Fig. 8(a). If $f < g$, P_i is the unique path in

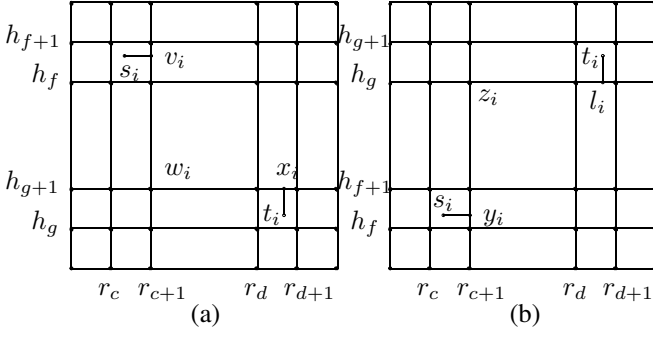


Fig. 8. Case 2 in the path choice algorithm (a) when $f > g$ and (b) when $f < g$.

$\mathcal{P}^*(s_i, y_i) \times \mathcal{P}^*(y_i, z_i) \times \mathcal{P}^*(z_i, l_i) \times \mathcal{P}^*(l_i, t_i)$ where $y_i = (r_{c+1}, s_{iy})$, $z_i = (r_{c+1}, h_g)$, and $l_i = (t_{ix}, h_g)$, as shown in Fig. 8(b).

Case 3) $c_1 > c_2$ and $d_1 \neq d_2$. We define unicast (s'_i, t'_i) for which $s'_i = t_i$ and $t'_i = s_i$. Then, by case 2), we can obtain a shortest path P'_i for (s'_i, t'_i) . In this case, we set $P_i = (P'_i)^R$.

B. Edge Use Distribution

As in Sec. III-B, we determine $r^1(e)$ for $e \in E$.

Case 1) $e = ((a, b), (a + 1, b))$, $e \in E_1$. Since $e \in E_1$, $r_p \leq a < r_{p+1}$ and $b = h_q$ for some $0 \leq p \leq k$ and $1 \leq q \leq t$, respectively. If $0 \leq s_{1x} < r_p$ and $s_{1y} \leq t_{1y}$, $e \in P_1$ if and only if $0 \leq s_{1y} \leq b$, $a + 1 \leq t_{1x} \leq m$, and $b \leq t_{1y} < h_{q+1}$, as shown in Fig. 9(a). If $0 \leq s_{1x} < r_p$ and $s_{1y} > t_{1y}$, $e \in P_1$ if and only if $b \leq s_{1y} \leq m$, $a + 1 \leq t_{1x} \leq m$, and $h_{q-1} \leq t_{1y} < b$, as shown in Fig. 9(b). If $r_p \leq s_{1x} \leq a$, $e \in P_1$ if and only if $s_{1y} = b$, $a + 1 \leq t_{1x} \leq m$, and $0 \leq t_{1y} \leq m$, as shown in Fig. 10(a). If $s_{1x} > a$, then $e \notin P_1$. Thus,

$$r^1(e) = \left[\frac{r_p(b+1)(m-a)(h_{q+1}-b)}{(m+1)^4} + \frac{r_p(m+1-b)(m-a)(b-h_{q-1})}{(m+1)^4} + \frac{(a+1-r_p)(m-a)(m+1)}{(m+1)^4} \right].$$

Case 2) $e = ((a, b), (a + 1, b))$ and $e \notin E_1$. Since $e \notin E_1$, $r_p \leq a < r_{p+1}$ and $h_q < b < h_{q+1}$ for some $0 \leq p \leq k$ and $0 \leq q \leq t$, respectively. If $0 \leq s_{1x} < r_p$, then $e \in P_1$ if and only if $s_{1y} = b$, $a + 1 \leq t_{1x} \leq m$, and $h_q \leq t_{1y} < h_{q+1}$, as shown in Fig. 10(b). If $r_p \leq s_{1x} \leq a$, then $e \in P_1$ if and only if $s_{1y} = b$, $a + 1 \leq t_{1x} \leq m$, and $0 \leq t_{1y} \leq m$. If $s_{1x} > a$, then $e \notin P_1$. Thus,

$$r^1(e) = \left[\frac{r_p(m-a)(h_{q+1}-h_q)}{(m+1)^4} + \frac{(a+1-r_p)(m-a)(m+1)}{(m+1)^4} \right].$$

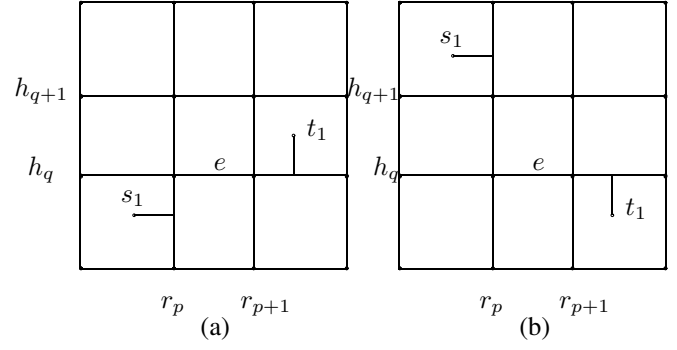


Fig. 9. Case 1 in the calculation of edge use distribution $r^1(e)$ when $0 \leq s_{1x} < r_p$. (a) $s_{1y} \leq t_{1y}$ and (b) $s_{1y} > t_{1y}$.

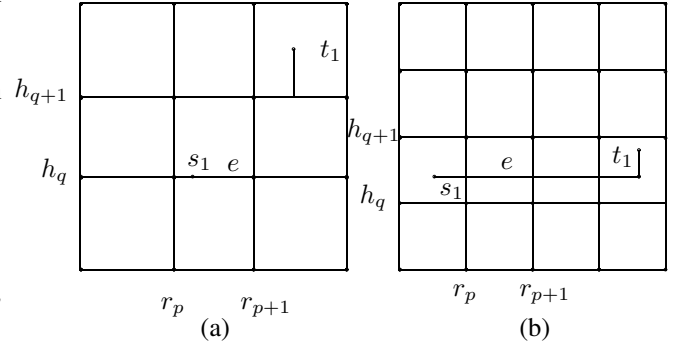


Fig. 10. (a) Case 1 in the calculation of edge use distribution $r^1(e)$ when $r_p \leq s_{1x} \leq a$. (b) Case 2 in the calculation of edge use distribution $r^1(e)$ when $0 \leq s_{1x} < r_p$.

Case 3) $e = ((a + 1, b), (a, b))$. We use $X(e)$ to denote the set of (s_1, t_1) such that $e \in P_1$. We show that there is an one to one correspondence between $X(e)$ and $X(e^R)$. Choose $0 \leq c, d \leq k$ and $0 \leq f, g \leq t$ so that $r_c \leq s_{1x} < r_{c+1}$, $r_d \leq t_{1x} < r_{d+1}$, $h_f \leq s_{1y} < h_{f+1}$, and $h_g \leq t_{1y} < h_{g+1}$. When $c = d$ or $f = g$, $e \in P_1$ if and only if $e^R \in P_1'$ for $(s'_1, t'_1) = ((t_{1x}, s_{1y}), (s_{1x}, t_{1y}))$. When $c \neq d$ and $f \neq g$, by the symmetry of our path choice algorithm, $e \in P_1$ if and only if $e^R \in P_1'$ for $(s'_1, t'_1) = (t_1, s_1)$. Thus, there exists an one to one correspondence between $X(e)$ and $X(e^R)$ and thus $|X(e)| = |X(e^R)|$. Then we can calculate $r^1(e) = r^1(e^R)$ from cases 1 and 2.

Case 4) $e = ((a, b), (a, b + 1))$, $e \in E_1$. Since $e \in E_1$, $a = r_p$ and $h_q \leq b < h_{q+1}$ for some $1 \leq p \leq k$ and $0 \leq q \leq t$, respectively. Choose $0 \leq c, d \leq k$ and $0 \leq f, g \leq t$ as we did in case 3. If $c = d$, then $e \in P_1$ if and only if $r_p \leq s_{1x} < r_{p+1}$, $0 \leq s_{1y} \leq b$, $t_{1x} = a$, and $b + 1 \leq t_{1y} \leq m$. If $f = g$ and $c \neq d$, then $e \in P_1$ if and only if $0 \leq s_{1x} < r_p$ or $r_{p+1} \leq s_{1x} \leq m$, $h_q \leq s_{1y} \leq b$, $t_{1x} = a$, and $b + 1 \leq t_{1y} < h_{q+1}$. If $f < g$ and $c < d$ and when $h_{q+1} \leq t_{1y} \leq m$, then $e \in P_1$ if and only if $r_{p-1} \leq s_{1x} < a$, $0 \leq s_{1y} \leq b$, and $a \leq t_{1x} \leq m$, as shown in Fig. 11(a). If

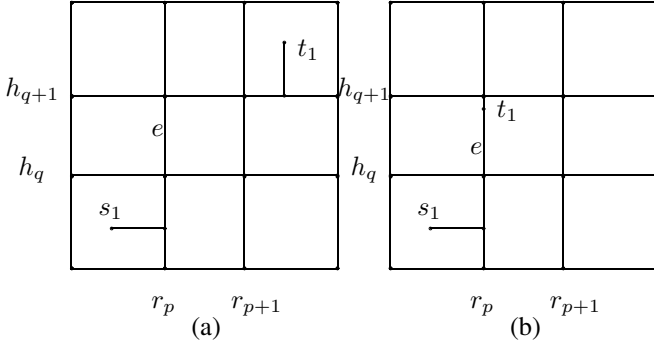


Fig. 11. Case 4 in the calculation of edge use distribution $r^1(e)$ when $c < d$ and $f < g$. (a) $h_{q+1} \leq t_{1y} \leq m$ and (b) $b+1 \leq t_{1y} < h_{q+1}$.

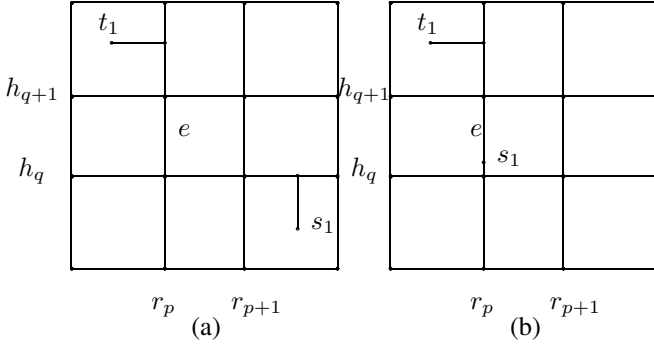


Fig. 12. Case 4 in the calculation of edge use distribution $r^1(e)$ when $c > d$ and $f < g$. (a) $0 \leq s_{1y} < h_q$ and (b) $h_q \leq s_{1y} \leq b$.

$f < g$ and $c < d$ and when $b+1 \leq t_{1y} < h_{q+1}$, then $e \in P_1$ if and only if $0 \leq s_{1x} < a$, $0 \leq s_{1y} < h_q$, and $t_{1x} = a$, as shown in Fig. 11(b). If $f < g$ and $c > d$ and when $0 \leq s_{1y} < h_q$, then $e \in P_1$ if and only if $a \leq s_{1x} \leq m$, $r_{p-1} \leq t_{1x} < a$, and $b+1 \leq t_{1y} \leq m$, as shown in Fig. 12(a). If $f < g$ and $c > d$ and when $h_q \leq s_{1y} \leq b$, then $e \in P_1$ if and only if $s_{1x} = a$, $0 \leq t_{1x} < a$, and $h_{q+1} \leq t_{1y} \leq m$, as shown in Fig. 12(b). If $f > g$, then $e \notin P_1$. Thus,

$$r^1(e) = \left[\frac{(r_{p+1} - r_p)(b+1)(m-b)}{(m+1)^4} + \frac{(m+1-r_{p+1}+r_p)(b+1-h_q)(h_{q+1}-b-1)}{(m+1)^4} + \frac{(m+1-h_{q+1})(a-r_{p-1})(b+1)(m+1-a)}{(m+1)^4} + \frac{(h_{q+1}-b-1)ah_q}{(m+1)^4} + \frac{h_q(m+1-a)(a-r_{p-1})(m-b)}{(m+1)^4} + \frac{(b+1-h_q)a(m+1-h_{q+1})}{(m+1)^4} \right].$$

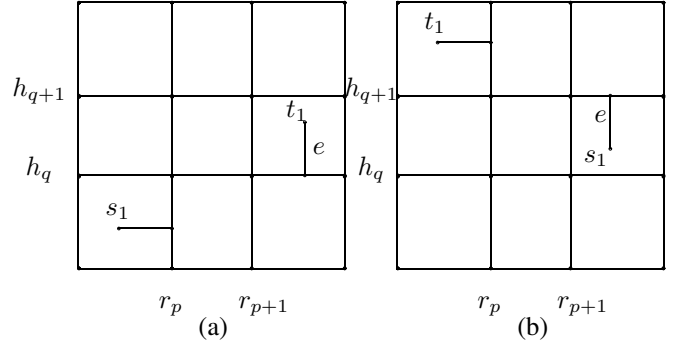


Fig. 13. Case 5 in the calculation of edge use distribution $r^1(e)$ when $c \neq d$ and $f \neq g$. (a) $c < d$ and (b) $c > d$.

Case 5) $e = ((a, b), (a, b+1))$ and $e \notin E_1$. Since $e \notin E_1$, $r_p < a < r_{p+1}$ and $h_q \leq b < h_{q+1}$ for some $0 \leq p \leq k$ and $0 \leq q \leq t$, respectively. If $c = d$, then $e \in P_1$ if and only if $r_p \leq s_{1x} < r_{p+1}$, $0 \leq s_{1y} \leq b$, $t_{1x} = a$, and $b+1 \leq t_{1y} \leq m$. If $f = g$ and $c \neq d$, then $e \in P_1$ if and only if $0 \leq s_{1x} < r_p$ or $r_{p+1} \leq s_{1x} \leq m$, $h_q \leq s_{1y} \leq b$, $t_{1x} = a$, $b+1 \leq t_{1x} < h_{q+1}$. If $c < d$ and $f \neq g$, then $e \in P_1$ if and only if $0 \leq s_{1x} < r_p$, $0 \leq s_{1y} < h_q$, $t_{1x} = a$, and $b+1 \leq t_{1y} < h_{q+1}$, as shown in Fig. 13(a). If $c > d$ and $f \neq g$, then $e \in P_1$ if and only if $s_{1x} = a$, $h_q \leq s_{1y} \leq b$, $0 \leq t_{1x} < r_p$, and $h_{q+1} \leq t_{1y} \leq m$, as shown in Fig. 13(b). Thus,

$$r^1(e) = \left[\frac{(r_{p+1} - r_p)(b+1)(m-b)}{(m+1)^4} + \frac{(m+1-r_{p+1}+r_p)(b+1-h_q)(h_{q+1}-b-1)}{(m+1)^4} + \frac{r_ph_q(h_{q+1}-b-1)}{(m+1)^4} + \frac{(b+1-h_q)r_p(m+1-h_{q+1})}{(m+1)^4} \right].$$

Case 6) $e = ((a, b+1), (a, b))$. As in case 3, we show that there exists one to one correspondence between $X(e)$ and $X(e^R)$. When $c = d$ or $f = g$, $e \in P_1$ if and only if $e^R \in P_1'$ for $(s_1', t_1') = ((s_{1x}, t_{1y}), (t_{1x}, s_{1y}))$. When $c \neq d$ and $f \neq g$, $e \in P_1$ if and only if $e^R \in P_1'$ for $(s_1', t_1') = (t_1, s_1)$. Then $X(e) = X(e^R)$ and we can calculate $r^1(e) = r^1(e^R)$ from cases 4 and 5.

C. Results

Let n unicasts $U = \{(s_1, t_1), \dots, (s_n, t_n)\}$ be chosen uniformly at random on the wireless rectangular grid G_m . Table 3 shows the optimal number of row and column reverse carpooling lines (t^*) and (k^*), and their optimal locations ($h_1^*, \dots, h_{t^*}^*$) and ($r_1^*, \dots, r_{k^*}^*$) for $m = 8$. To obtain the optimal reverse carpooling line placement in this case, we search over all possible choices of (h_1, \dots, h_t) and (r_1, \dots, r_k) for $1 \leq t, k \leq m$ and choose the one that minimizes the

TABLE III
OPTIMAL ROW AND COLUMN REVERSE CARPOOLING LINES PLACEMENT
ON THE G_8 .

n	t^*	(h_1^*, \dots, h_t^*)	k^*	(r_1^*, \dots, r_k^*)
$n < 20$	3	(2,4,6)	2	(3,5)
$20 \leq n$	3	(2,4,6)	2	(2,5)

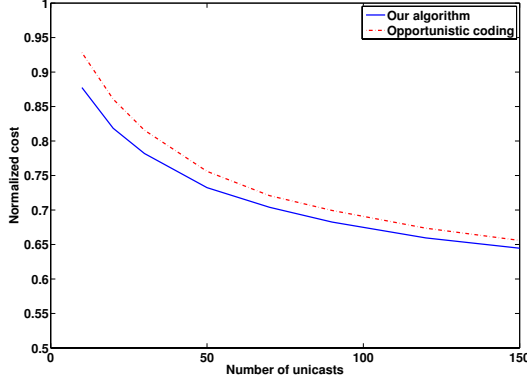


Fig. 14. Normalized cost on G_8 .

expected network cost. Since we cannot decompose the optimization problem into independent subproblems in this case, we cannot apply the algorithm from Sec III.

Fig. 14 plots the normalized cost as a function of the number of unicasts. The normalized cost decreases as the number of unicast sessions increases. Corresponding normalized cost when pure opportunistic coding is employed is also included. Our algorithm yields as much as 7% improvement over pure opportunistic coding when $n < 30$. Similar to the result presented in Sec. III-D, the percentage improvement over opportunistic coding decreases as n increases.

V. DISCUSSION

In this paper, we present a distributed strategy for reducing the expected power consumption for multiple unicasts in a network coded wireless network. We use the rectangular grid as a simple network model and apply a simple network coding strategy called “reverse carpooling,” which uses only XOR and forwarding operations. Our strategy is to attempt to increase the number of coding opportunities by designating “reverse carpooling routes.” Each individual unicast chooses a route from its source to its destination independently but in a manner that maximizes the fraction of the paths spent on the reverse carpooling lines without increasing individual path lengths. Intermediate nodes apply reverse carpooling opportunistically along these routes. This approach increases the reverse carpooling opportunities of an opportunistic network code without requiring central coordination.

We propose distributed route choice algorithms for row model and row and column model respectively, and analyze the edge use distribution of our algorithms. Then we can optimize the reverse carpooling line placement to minimize the resulting expected cost. When all reverse carpooling lines

are rows, we present recursive algorithm that optimizes the line choice in time $O(qm^6 + n^2m^6)$ where m is the grid size, n is the number of unicasts, and $2^{q+1} - 2$ is the number of reverse carpooling lines. This algorithm yields as much as 7% improvement over pure opportunistic coding when $n < 40$. When reverse carpooling lines include both rows and columns, we optimize the line choice by brute force search and our strategy also yields 7% improvement over pure opportunistic coding when $n < 30$.

APPENDIX

Lemma 1: When $U = \{(s_1, t_1), \dots, (s_n, t_n)\}$ is chosen i.i.d. from the uniform distribution on G_m ,

$$ED(U) = \frac{2m(m+2)n}{3(m+1)}.$$

Proof:

$$\begin{aligned} ED(U) &= nEd(s_1, t_1) \\ &= \frac{n \sum_{0 \leq s_{1x}, s_{1y}, t_{1x}, t_{1y} \leq m} d(s_1, t_1)}{(m+1)^4} \\ &= \frac{n}{2(m+1)^4} \sum_{s_{1x}, s_{1y}} [(s_{1x}+1)(s_{1y}+1)(s_{1x}+s_{1y}) \\ &\quad + (s_{1x}+1)(s_{1x}+m-s_{1y}+1)(m-s_{1y}) \\ &\quad + (s_{1y}+1)(s_{1y}+m-s_{1x}+1)(m-s_{1x}) \\ &\quad + (m-s_{1x})(m-s_{1y})(2m-s_{1x}-s_{1y}+2)] \\ &= \frac{2m(m+2)n}{3(m+1)} \end{aligned}$$

ACKNOWLEDGEMENTS

This work has been supported in part by the Defense Advanced Research Projects Agency (DARPA) under Contract No. W911NF-07-1-0029, and by Caltech’s Lee Center for Advanced Networking.

REFERENCES

- [1] R. Ahlswede, N. Cai, S.-Y.R. Li, and R. W. Yeung, “Network information flow,” *IEEE Trans. Inform. Theory*, vol 46, no. 4, pp. 1204-1216, July 2000.
- [2] Y. Wu, P.A. Chou, and S.-Y. Kung, “Minimum-energy multicast in mobile ad-hoc networks using network coding,” *IEEE Trans. Commun.*, vol 53, no. 11, pp. 1906-1918, Nov. 2005.
- [3] Y. Wu, P.A. Chou, and S.-Y. Kung, “Information exchange in wireless networks with network coding and physical layer broadcast,” *Technical Report MSR-TR-2004-78*, Microsoft Research, Redmond, WA, Aug. 2004.
- [4] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard and J. Crowcroft, “XORs in the Air: Practical Wireless Network Coding,” in *Proc. of ACM SIGCOMM 2006*.
- [5] M. Effros, T. Ho, and S. Kim, “A Tiling Approach to Network Code Design for Wireless Networks,” in *Proc. of Information Theory Workshop (ITW)*, Punta del Este, Uruguay, Mar. 2006, pp. 62-66.
- [6] S. Kim, M. Effros, and T. Ho, “On Low-Power Multiple Unicast Network Coding Over a Wireless Triangular Grid”, 45th Allerton Annual Conference on Communication, Control, and Computing, 2007.
- [7] J. Widmer, C. Fragouli, and J.-Y. L. Boudec, “Low-complexity energy-efficient broadcasting in wireless ad-hoc networks using network coding,” in *Proc. of Workshop on Network Coding, Theory, and Applications (NetCod 2005)*, Riva del Garda, Italy, Apr. 2005.
- [8] A. Tucker, “Applied Combinatorics”, Hoboken, NJ: John Wiley and Sons, 1994.